RLY

UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/873,730	06/02/2001	Gerald Neufeld	4906.P021	4760

7590	03/25/2004

Daniel M. DeVos
Blakely, Sokoloff, Taylor & Zafman LLP
Seventh Floor
12400 Wilshire Boulevard
Los Angeles, CA 90025-1030

EXAMINER
PHAM, HUNG Q

ART UNIT	PAPER NUMBER
2172	4

DATE MAILED: 03/25/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	09/873,730	NEUFELD ET AL.
	Examiner	Art Unit
	HUNG Q PHAM	2172

-- *The MAILING DATE of this communication appears on the cover sheet with the correspondence address* --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) FROM
THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed
  after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any
  earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☐ Responsive to communication(s) filed on _____.

2a)☐ This action is **FINAL**.   2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is
closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) _1-53_ is/are pending in the application.

     4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) _1-53_ is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on _02 June 2001_ is/are: a)☒ accepted or b)☐ objected to by the Examiner.

     Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

     Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. §§ 119 and 120

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

     a)☐ All b)☐ Some * c)☐ None of:

         1.☐ Certified copies of the priority documents have been received.

         2.☐ Certified copies of the priority documents have been received in Application No. _____.

         3.☐ Copies of the certified copies of the priority documents have been received in this National Stage
application from the International Bureau (PCT Rule 17.2(a)).

     * See the attached detailed Office action for a list of the certified copies not received.

13)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application)
since a specific reference was included in the first sentence of the specification or in an Application Data Sheet.
37 CFR 1.78.

     a) ☐ The translation of the foreign language provisional application has been received.

14)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121 since a specific
reference was included in the first sentence of the specification or in an Application Data Sheet. 37 CFR 1.78.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)   4) ☐ Interview Summary (PTO-413) Paper No(s). _____.

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)   5) ☐ Notice of Informal Patent Application (PTO-152)

3) ☒ Information Disclosure Statement(s) (PTO-1449) Paper No(s) _2 & 3_.   6) ☐ Other: .

DETAILED ACTION

Claim Rejections - 35 USC § 112

1.   The following is a quotation of the first paragraph of 35 U.S.C. 112:

The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

Claims 1-2, 7-8, 24, 30, 34, 36-38 and 43-44 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter, which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention. As in claims 1 and 37, the steps of *executing the operation idempotently with a network resource process, moving the record into an atomic database* as in claims 2 and 38, *performing the sequence of operations as an atomic transaction* as in claims 7 and 43, *each of the sequence of operations is performed idempotently* as in claims 8 and 44, *storing a first and second operation as an atomic transaction; performing the first and second operation idempotently with a set of network resource process* as in claim 24, *executing each of the set of operations idempotently* as in claim 30, *processing the plurality of operations as an atomic transaction* as in claim 34, *the second interface to receive a second plurality of operations from a second user* as in claim 36, were not described in the specification.

Claim Rejections - 35 USC § 102

2.   The following is a quotation of the appropriate paragraphs of 35

U.S.C. 102 that form the basis for the rejections under this section made in this

Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in a patent granted on an application for patent by another filed in the United States before the invention thereof by the applicant for patent, or on an international application by another who has fulfilled the requirements of paragraphs (1), (2), and (4) of section 371(c) of this title before the invention thereof by the applicant for patent.

The changes made to 35 U.S.C. 102(e) by the American Inventors

Protection Act of 1999 (AIPA) and the Intellectual Property and High Technology

Technical Amendments Act of 2002 do not apply when the reference is a U.S.

patent resulting directly or indirectly from an international application filed before

November 29, 2000. Therefore, the prior art date of the reference is determined

under 35 U.S.C. 102(e) prior to the amendment by the AIPA (pre-AIPA 35 U.S.C.

102(e)).

Claims 1-5, 7-10, 12-16, 18-21, 23-26, 29-41, 43-46, 48-51 and 53 are

rejected under 35 U.S.C. 102(e) as being anticipated by Traversat et al.

[USP 6,115,715].

Regarding to claims 1 and 37, Traversat teaches a method for updating

and managing a configuration database. As shown in FIG. 5, at step 502, the

transaction of adding a new printer to a client will attempt to lock the appropriate

node in a sub-tree of the client JSD schema. At step 504 the system determines

whether the transaction is a blocking or unblocking transaction (Col. 8, Lines 3-8). If the transaction is a blocking transaction, it will wait until the lock is released by the current transaction (Col. 8, Lines 8-10). Any transactions waiting to obtain a lock, regardless of whether exclusive or shared, registers itself with the event manager. The central event manager can then determine which transactions are waiting for the node or nodes that were released (Col. 9, Lines 32-36). As shown in FIG. 8 is the event queue for the transaction identified by a transaction handle (Col. 10, Line 64-Col. 11, Line 2). Other mechanisms can be used to keep track of previous activity such as a relational table or other type of database from which activity data can be stored in a chronological manner (Col. 9, Lines 55-58). Returning to FIG. 5 at step 508, if the update is successful, the configuration database as *a network resource process* is updated. If there is a failure and an update is not successful for any reason, control goes to step 514 where the transaction is aborted. This can happen if a user adds an incompatible printer or attempt to install an incompatible software program to the computer, or attempts to add a device already connected to his computer. In the abort phase, the goal is to return the configuration database to a consistent state (Col. 9, Lines 42-58). As seen, the step of placing the transaction in a queue and a database from which activity data can be stored in a chronological manner as *storing an operation*. If an error occurs, the transactions are executed in order to return the configuration database to a consistent state indicates the step of *executing the operation idempotently with a network resource process*.

Regarding to claims 2 and 38, Traversat teaches all the claimed subject matters as discussed in claims 1 and 37, Traversat further discloses the step of *storing the operation in a log as a record; receiving a commit command, and moving the record into an atomic database* (Col. 9, Lines 52-58).

Regarding to claims 3 and 39, Traversat teaches all the claimed subject matters as discussed in claims 1 and 37, Traversat further discloses the steps of *receiving the operation; performing lock contention handling for the operation; storing the operation if a lock contention is not detected; and generating a lock contention notification if the lock contention is detected for the operation* (Col. 8, Line 3-Col. 9, Line 65).

Regarding to claims 4 and 40, Traversat teaches all the claimed subject matters as discussed in claims 1 and 37, Traversat further discloses *the operation is one of a sequence of operations comprising an atomic transaction* (FIG. 8).

Regarding to claims 5 and 41, Traversat teaches all the claimed subject matters as discussed in claims 1 and 37, Traversat further discloses the steps of *receiving the operation from a first user; and receiving a second operation from a second user* (FIG. 8).

Regarding to claims 7 and 43, Traversat teaches a method for updating and managing a configuration database. As shown in FIG. 5, at step 502, the

transaction of adding a new printer to a client will attempt to lock the appropriate

node in a sub-tree of the client JSD schema. At step 504 the system determines

whether the transaction is a blocking or unblocking transaction (Col. 8, Lines 3-

8). If the transaction is a blocking transaction, it will wait until the lock is released

by the current transaction (Col. 8, Lines 8-10). Any transactions waiting to obtain

a lock, regardless of whether exclusive or shared, registers itself with the event

manager. The central event manager can then determine which transactions are

waiting for the node or nodes that were released (Col. 9, Lines 32-36). As shown

in FIG. 8 is the event queue for the transaction identified by a transaction handle

(Col. 10, Line 64-Col. 11, Line 2). Other mechanisms can be used to keep track

of previous activity such as a relational table or other type of database from

which activity data can be stored in a chronological manner (Col. 9, Lines 55-58).

Once an event queue for the transaction is available, all state data relating to a

specific update that is necessary to undo the update if necessary is stored as an

entry in the event queue. By doing this incrementally for each specific update

performed in a transaction, the system can return the configuration database to

its original state before the transaction started (Col. 10, Lines 18-29). Returning

to FIG. 5 at step 508, if the update is successful, the configuration database is

updated. If there is a failure and an update is not successful for any reason,

control goes to step 514 where the transaction is aborted. This can happen if a

user adds an incompatible printer or attempt to install an incompatible software

program to the computer, or attempts to add a device already connected to his

computer. In the abort phase, the goal is to return the configuration database to a

consistent state (Col. 9, Lines 42-58). As seen, the step of placing the

transactions in a queue, and a database from which activity data can be stored in

a chronological manner as *storing a sequence of operations*. The queue is

executed, and if the transaction in the queue is aborted, the abort phrase is

provided to ensure the return of the configuration database to its state prior to

initiation of the operation, in different words, this indicates the step of *performing*

the sequence of operations as an atomic transaction.

Regarding to claims 8 and 44, Traversat teaches all the claimed subject

matters as discussed in claims 7 and 43, Traversat further discloses *each of the*

sequence of operations is performed idempotently (Col. 9, Lines 42-58).

Regarding to claims 9 and 45, Traversat teaches all the claimed subject

matters as discussed in claims 7 and 43, Traversat further discloses the steps of

performing lock contention handling for each of the sequence of the operation; storing

the sequence of the operation if a lock contention is not detected; generating a lock

contention notification if the lock contention is detected (Col. 8, Line 3-Col. 9, Line

65).

Regarding to claims 10 and 46, Traversat teaches all the claimed subject

matters as discussed in claims 7 and 43, Traversat further discloses *a sequence of*

operations is received from a first user and a second sequence of operations is received

from a second user (FIG. 8).

Regarding to claim 12, Traversat teaches a method for updating and

managing a configuration database. As shown in FIG. 5, at step 502, the

transaction of adding a new printer to a client will attempt to lock the appropriate

node in a sub-tree of the client JSD schema. At step 504 the system determines

whether the transaction is a blocking or unblocking transaction (Col. 8, Lines 3-

8). If the transaction is a blocking transaction, it will wait until the lock is released

by the current transaction (Col. 8, Lines 8-10). Any transactions waiting to obtain

a lock, regardless of whether exclusive or shared, registers itself with the event

manager. The central event manager can then determine which transactions are

waiting for the node or nodes that were released (Col. 9, Lines 32-36). As shown

in FIG. 8 is the event queue of the transaction identified by a transaction handle

(Col. 10, Line 64-Col. 11, Line 2). Other mechanisms can be used to keep track

of previous activity such as a relational table or other type of database from

which activity data can be stored in a chronological manner (Col. 9, Lines 55-58).

As seen, the step of placing the transactions in a queue and a database from

which activity data can be stored in a chronological manner as *storing an*

operation in an atomic database. As shown in FIG. 5, at step 510 the update or

updates making up the single transaction performed at step 508 are committed

and a notice that the transaction has been committed is broadcast to all threads

waiting on the nodes that were locked. The event manager is notified once a lock

is released by posting the release to the event manager. The central event

manager then determines which transactions are waiting for the node or nodes

that were released for notification (Col. 9, Lines 7-41). As seen, the transaction in

the queue begins its performance in response to a commit command of previous

transaction to update the configuration database as network resource process. In

other words, the technique as discussed indicates the step of *performing the*

operation with a network resource process in response to a commit command.

Regarding to claim 13, Traversat teaches all the claim subject matters as

discussed in claim 12, Traversat further discloses *the operation is performed*

idempotently (Col. 9, Lines 42-58).

Regarding to claim 14, Traversat teaches all the claim subject matters as

discussed in claim 12, Traversat further discloses *the operation is one of a*

sequence of operations comprising a transaction (FIG. 8).

Regarding to claim 15, Traversat teaches all the claim subject matters as

discussed in claim 12, Traversat further discloses the steps of *performing lock*

contention handling for the operation; storing the operation in the atomic database if a

lock contention is not detected; and generating a notification of lock contention if the

lock contention is detected (Col. 8, Line 3-Col. 9, Line 65).

Regarding to claim 16, Traversat teaches all the claim subject matters as

discussed in claim 12, Traversat further discloses *the operation is received from a*

first user and the second operation is received from a second user (FIG. 8).

Regarding to claims 18 and 48, Traversat teaches a method for updating

and managing a configuration database. As shown in FIG. 5, at step 502, the

transaction of adding a new printer to a client will attempt to lock the appropriate

node in a sub-tree of the client JSD schema (Col. 8, Lines 3-5) as the step of

receiving an operation. If the attempt to lock the appropriate parent node or leaf

node fails, control goes to step 504 (Col. 8, Lines 5-7) as the step of *determining*

if a lock contention exists for a record corresponding to the operation. At step 504 the

system determines whether the transaction is a blocking or unblocking

transaction (Col. 8, Lines 7-8). If the transaction is blocking, control returns to

step 502 where the transaction attempts to acquire a lock on the desired entry,

this time after waiting in a queue and being notified by an event manager (Col. 8,

Lines 35-38). As seen, the transaction is stored in a queue and being notified

when a lock exist for the node. In other words, this performs the step of

generating a notification of the lock contention if a lock contention does exist for the

record.

Regarding to claims 19 and 49, Traversat teaches all the claim subject

matters as discussed in claims 18 and 48, Traversat further discloses *the*

operation is one of a sequence of operations comprising an atomic transaction (FIG.

8).

Regarding to claims 20 and 50, Traversat teaches all the claim subject matters as discussed in claims 18 and 48, Traversat further discloses *the operation is performed idempotently with a network resource process* (Col. 9, Lines 42-58).

Regarding to claims 21 and 51, Traversat teaches all the claim subject matters as discussed in claims 18 and 48, Traversat further discloses *the operation is received from a first user and the second operation is received from a second user* (FIG. 8).

Regarding to claims 23 and 53, Traversat teaches all the claim subject matters as discussed in claims 18 and 48, Traversat further discloses the step of *storing the operation if the lock contention does not exist* (FIG. 8).

Regarding to claim 24, Traversat teaches a method for updating and managing a configuration database. As shown in FIG. 5, at step 502, the transaction of adding a new printer to a client will attempt to lock the appropriate node in a sub-tree of the client JSD schema. At step 504 the system determines whether the transaction is a blocking or unblocking transaction (Col. 8, Lines 3-8). If the transaction is a blocking transaction, it will wait until the lock is released by the current transaction (Col. 8, Lines 8-10). Any transactions waiting to obtain a lock, regardless of whether exclusive or shared, registers itself with the event manager. The central event manager can then determine which transactions are

waiting for the node or nodes that were released (Col. 9, Lines 32-36). As shown

in FIG. 8 is the event queue for the transaction identified by a transaction handle

(Col. 10, Line 64-Col. 11, Line 2). Other mechanisms can be used to keep track

of previous activity such as a relational table or other type of database from

which activity data can be stored in a chronological manner (Col. 9, Lines 55-58).

Once an event queue for the transaction is available, all state data relating to a

specific update that is necessary to undo the update if necessary is stored as an

entry in the event queue. By doing this incrementally for each specific update

performed in a transaction, the system can return the configuration database to

its original state before the transaction started (Col. 10, Lines 18-29). Returning

to FIG. 5 at step 508, if the update is successful, the configuration database is

updated. If there is a failure and an update is not successful for any reason,

control goes to step 514 where the transaction is aborted. This can happen if a

user adds an incompatible printer or attempt to install an incompatible software

program to the computer, or attempts to add a device already connected to his

computer. In the abort phase, the goal is to return the configuration database to a

consistent state (Col. 9, Lines 42-58). As seen, the step of placing the aborted

transactions in a queue, and a database from which activity data can be stored in

a chronological manner as *storing a first and a second operation as an atomic

transaction*. Returning to FIG. 5 at step 508, if there is a failure and an update is

not successful for any reason, control goes to step 514 where the transaction is

aborted. This can happen if a user adds an incompatible printer or attempt to

install an incompatible software program to the computer, or attempts to add a

device already connected to his computer. In the abort phase, the goal is to

return the configuration database as *a set of network resource process* to a

consistent state (Col. 9, Lines 42-58). As seen, if an error occurs, the

transactions are executed in order to return the configuration database to a

consistent state indicates the step of *performing the first and second operation*

idempotently with a set of network resource process.

Regarding to claim 25, Traversat teaches all the claim subject matters as

discussed in claim 24, Traversat further discloses the step of *performing lock*

contention handling for the first and second operation; storing the first and second

operation of the atomic transaction in a log if a lock contention is not detected; and

generating a lock contention notification if the lock contention is detected (Col. 8,

Line 3-Col. 9, Line 65).

Regarding to claim 26, Traversat teaches all the claim subject matters as

discussed in claim 24, Traversat further discloses the step of *storing the first and*

second operation of the atomic transaction in a log; receiving a commit command for

the atomic transaction; indicating the atomic transaction as committed; and storing the

atomic transaction in an atomic database (Col. 9, Lines 7-58).

Regarding to claim 29, Traversat teaches a system and method for

updating and managing a configuration database. As shown in FIG. 5, at step

502, the transaction of adding a new printer to a client will attempt to lock the

appropriate node in a sub-tree of the client JSD schema. At step 504 the system

determines whether the transaction is a blocking or unblocking transaction (Col.

8, Lines 3-8). If the transaction is a blocking transaction, it will wait until the lock

is released by the current transaction (Col. 8, Lines 8-10). Any transactions

waiting to obtain a lock, regardless of whether exclusive or shared, registers itself

with the event manager. The central event manager can then determine which

transactions are waiting for the node or nodes that were released (Col. 9, Lines

32-36). As shown in FIG. 8 is the event queue of the transaction identified by a

transaction handle (Col. 10, Line 64-Col. 11, Line 2). Other mechanisms can be

used to keep track of previous activity such as a relational table or other type of

database from which activity data can be stored in a chronological manner (Col.

9, Lines 55-58). As seen, the step of placing the aborted transactions in a queue

and a database from which activity data can be stored in a chronological manner

as *storing the set of atomic transactions*. Returning to FIG. 5 at step 508, if the

update is successful, the configuration database is updated. If there is a failure

and an update is not successful for any reason, control goes to step 514 where

the transaction is aborted. This can happen if a user adds an incompatible printer

or attempt to install an incompatible software program to the computer, or

attempts to add a device already connected to his computer. In the abort phase,

the goal is to return the configuration database to a consistent state (Col. 9, Lines

42-58). As seen, the queue is executed, and if the transaction in the queue is

aborted, the abort phrase is provided to ensure the return of the configuration

database to its state prior to initiation of the operation, in different words, this

indicates the step of *executing a set of atomic transactions*.

Regarding to claim 30, Traversat teaches all the claim subject matters as

discussed in claim 29, Traversat further discloses the step of *executing each of the*

set of operations idempotently (Col. 9, Lines 42-58).

Regarding to claim 31, Traversat teaches all the claim subject matters as

discussed in claim 29, Traversat further discloses the step of *forming lock*

contention handling for the set of atomic transactions; and generating the lock

contention notification if the lock contention is detected (Col. 8, Line 3-Col. 9, Line

65).

Regarding to claim 32, Traversat teaches all the claim subject matters as

discussed in claim 29, Traversat further discloses *a set of interfaces coupled to the*

processor, each of the set if interfaces to receive at least one of the set of atomic

transactions, each of the set of interfaces corresponding to a different user (FIG. 1).

Regarding to claim 33, Traversat teaches a network element comprising:

an interface to receive a plurality of operations from a user; a configuration manager

coupled to the interface, the configuration manager to process the plurality of

operations; and an atomic database coupled to the configuration manager, the atomic

database to store the plurality of operations as a transaction (FIG. 9, Col. 11, Line 43-
Col. 12, Line 64; Col. 9, Lines 54-58).

Regarding to claim 34, Traversat teaches all the claim subject matters as
discussed in claim 33, Traversat further discloses the step of *processing the
plurality of operations as an atomic transaction* (Col. 9, Lines 42-58).

Regarding to claim 35, Traversat teaches all the claim subject matters as
discussed in claim 33, Traversat further discloses *the atomic database to detect lock
contention; the configuration manager to generate a notification of the lock contention
detected by the atomic database; and the interface to display a message corresponding
to the notification generated by the configuration manger* (Col. 8, Line 3-Col. 9, Line
58).

Regarding to claim 36, Traversat teaches all the claim subject matters as
discussed in claim 33, Traversat further discloses *a second interface to receive a
second plurality of operations from a second user* (Col. 12, Line 65-Col. 13, Line 4).

Claim Rejections - 35 USC § 103

3.   The following is a quotation of 35 U.S.C. 103(a) which forms the

basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described
> as set forth in section 102 of this title, if the differences between the subject matter sought to
> be patented and the prior art are such that the subject matter as a whole would have been
> obvious at the time the invention was made to a person having ordinary skill in the art to which

said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

Claims 6, 11, 17, 22, 27-28, 42, 47 and 52 are rejected under 35 U.S.C. 103(a) as being unpatentable over Traversat et al. [USP 6,115,715].

Regarding to claims 6, 11, 17, 22, 42, 47 and 52, Traversat teaches all the claimed subject matters as discussed in claims 1, 7, 12, 18, 37, 43 and 48, but does not explicitly teach *the operation is received from a first user concurrently with a second operation received from a second user*. However, as disclosed by Traversat, any transactions waiting to obtain a lock, regardless of whether exclusive or shared, registers itself with the event manager. The central event manager can then determine which transactions are waiting for the node or nodes that were released (Col. 9, Lines 32-36). Thus, if based on the step of registering, the event manager could solve the problem of two concurrent

operations. It would have been obvious for one of ordinary skill in the art at the time the invention was made to modify the Traversat technique in order to access a configuration database.

Regarding to claim 27, Traversat teaches all the claim subject matters as discussed in claim 24, Traversat does not explicitly teach *the first and second operation are received from a first user and a third operation of a second transaction is received from a second user*. However, as shown in FIG. 8 is the event queue, a transaction to attach a new printer to a computer, the device and interface namespace are modified (Col. 7, Lines 50-52) as the first and second operation to update the configuration database, obviously, another update transaction from another user could be related to the device or interface namespace as a third operation of a second transaction will cause the transaction has to be queued. Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to modify the Traversat technique to have three operations from different users in order to manage the configuration database.

Regarding to claim 28, Traversat teaches all the claim subject matters as discussed in claim 24, but does not explicitly disclose *the second operation is received from a first user concurrently with a third operation received from a second user*. However, as disclosed by Traversat, any transactions waiting to obtain a lock, regardless of whether exclusive or shared, registers itself with the event manager. The central event manager can then determine which transactions are

waiting for the node or nodes that were released (Col. 9, Lines 32-36). Thus, if based on the step of registering, the event manager could solve the problem of two concurrent operations. It would have been obvious for one of ordinary skill in the art at the time the invention was made to modify the Traversat technique in order to access a configuration database.

Conclusion

4.   Any inquiry concerning this communication or earlier communications from the examiner should be directed to HUNG Q PHAM whose telephone number is 703-605-4242. The examiner can normally be reached on Monday-Friday. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, JOHN E BREENE can be reached on 703-305-9790. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Examiner Hung Pham
March 15, 2004

SHAHID ALAM
PRIMARY EXAMINER